Simulation of the National Airspace System for Safety Analysis

Annual Performance Report For the Period
March 1, 2001 – February 28, 2002

PIs: Drs. Amy Pritchett and Dave Goldsman
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta GA 30332-0205
(Tel) 404-894-0199
(Fax) 404-894-2301
Amy.Pritchett@isye.gatech.edu

Grant NAG 2-1291
Technical Monitor: Dr. Irv. Statler

Work started on this project on January 1, 1999, the first year of the grant. Following the outline of the grant proposal, a simulation architecture has been established which can incorporate the variety of types of models needed to accurately simulate national airspace dynamics. For the sake of efficiency, this architecture was based on an established single-aircraft flight simulator, the Reconfigurable Flight Simulator (RFS), already developed at Georgia Tech. Likewise, in the first year substantive changes and additions were made to the RFS to convert it into a simulation of the National Airspace System, with the flexibility to incorporate many types of models: aircraft models; controller models; airspace configuration generators; discrete event generators; imbedded statistical functions; and display and data outputs. The architecture has been developed with the capability to accept any models of these types; due to its object-oriented structure, individual simulator components can be added and removed during run-time, and can be compiled separately. Simulation objects from other projects should be easy to convert to meet architecture requirements, with the intent that both this project may now be able to incorporate established simulation components from other projects, and that other projects may easily use this simulation without significant time investment.

In year two of the project, several agent models were created, including:

- Point-mass aircraft models which follow waypoints.
- A Random-Aircraft-Generator, which can place aircraft at departure points and airspace entries according to pre-specified probability distributions for arrival rate and speed.
- Generic controller models, which control aircraft within a defined airspace sector, giving them steering commands to meet spacing constraints, and then 'pass-off' aircraft to neighbouring controllers upon exit from their airspace. These models, at this time, follow normative models of behavior; however, their structure allows for more elaborate behavioral models to be implemented if desired in the future.
- Error-Prone Controller Models, which add to the generic controller model the ability to pre-specify omissions of, or deviations from, actions required by normative behavior. At this time, for example, a model is being tested in which the controller 'forgets' about one aircraft, thereby not giving it steering commands, allowing for examination of the effect such an error – if not caught – has on the national airspace system throughout time. For Monte Carlo simulation runs, these errors are triggered at random times involving random controllers and random aircraft.
- Imbedded statistical functions, which can monitor air traffic flow specifications and safety requirements, and provide, at the end of the simulation run, documentation of the timing and frequency of events, average and variance of continuous measures, and confidence intervals for these statistics.

Other functions, such as hypothesis testing functions, are also under development. For example, we now keep a running confidence interval of the probability of an incident (poor aircraft separation) occurring; this confidence interval is updated after each aircraft arrives at the gate.

- Data and display output features, which provide visual reference of object behaviour during simulation runs and testing of new components. Because of the many different types of behaviours captured within the simulation, several types of output are given – data intensive file generation; an 'Aircraft Situation Display (ASD)' for monitoring the trajectories of the aircraft; and text screens for monitoring the discrete events generated by some of the modules listed above, as well as events detected by the statistical functions.

In the period covering this report, several activities have been completed. The first was to improve the 'State Updater' object so that it is a more robust and easy to use facet of simulation, suitable for wide-spread use. One specific capability we added to this object is to make it capable of re-synchronizing a few specified modules within the simulation at a time; with this ability, resynchronizations will not be blindly applied to the entire simulation. This effort required substantial re-writing of the 'State Updater' object; it required careful definition of when resynchronizations are called and on what basis.

Second, the concept of resynchronization is so novel that the timing of resynchronizations is not well understood, and currently is being implemented in an ad hoc, overly-conservative manner, with resulting loss of computational efficiency. Therefore, we developed a structured mechanism for establishing when the simulation should be re-synchronized. This mechanism includes both theory and guidelines communicable to the simulation community, as well as specific tools within the simulation architecture capable of better projecting when resynchronizations are necessary. Specifically, we implemented a neural network architecture for monitoring the timing of interactions between agents within the simulation. Throughout simulation runs, this architecture trains itself on the emergent dynamics within the simulation and learns to predict when interactions should occur. Once it is trained, this network provides a computationally efficient mechanisms for predicting when interactions will occur that is tuned to the particular configuration and scenario of the immediate simulation.

Third, our simulation developments have been, and continued to be, used by other researchers within NASA's Aviation Safety Program. To assist in their efforts, we have continued to provide, within the personal services time covered by the grant, documentation and support of our simulation software. Specifically the development included:

- Implementation of HLA RTI in RFS to support linking our simulation to the MIDAS framework used by other researchers within the aviation safety program;
- Working with ATAC to develop the enroute sector scenario involving air-traffic controller agents, pilot agents, aircraft, communications, and surveillance agents. Our developments included the communication and surveillance agents, with assistance to didn't the other agents.
- Development of a measurement management object capable of assessing interaction times between agents.